Indonesian Journal of Computer Science
ISSN 2302-4364 (*print*) dan 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam, No. 1, Padang, Indonesia, Telp. (0751) 7056199, 7058325
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

Support Vector Machine untuk Sentiment Analysis Bakal Calon Presiden Republik Indonesia 2024

Boby Pranata[1*], Susanti[2], Susi Erlinda[3], Hadi Asnal[4]
boyprant@gmail.com[1], susanti@sar.ac.id[2], susierlinda@sar.ac.id[3], hadiasnal@sar.ac.id[4]
[1,2,3,4]STMIK Amik Riau

Informasi Artikel

Diterima : 11 Jun 2023
Direview : 20 Jun 2023
Disetujui : 30 Jun 2023

Kata Kunci

Support Vector Machine, Bakal Calon Presiden, VADER Lexicon, Sentimen

Abstrak

Survei Lingkaran Survei Indonesia dan Indo Barometer menempatkan 3 figur bakal calon presiden tertinggi, figur tersebut adalah Ganjar Pranowo, Anies Baswedan, dan Prabowo Subianto. Tiga tokoh yang dihasilkan dari survei tersebut merupakan figur terkenal. Banyak berita yang membicarakan tokoh tersebut, begitu juga dengan masyarakat. Masyarakat baik yang mendukung ataupun tidak, banyak membicarakan pada media sosial. Salah satu media sosial yang digunakan untuk beropini figur tersebut adalah media sosial Twitter. Dengan sentimen terhadap figur bakal calon presiden, dapat mengetahui bagaimana masyarakat menyikapi terhadap figur tersebut, trend sentimen terhadap figur, dan kata-kata yang menjadi sentimen. Sentimen-sentimen yang diberikan oleh pengguna Twitter tersebut banyak berupa opini ataupun berita yang jumlahnya sangat banyak di Twitter. Banyak teknik otomasi yang digunakan untuk melakukan analisis sentimen seperti pendekatan berbasis leksikon, menggunakan machine learning, atau gabungan keduanya. Algoritma machine learning yang digunakan beragam salah satunya Support Vector Machine (SVM). Hasil pelabelan menggunakan VADER Lexicon didapatkan hasil sentimen positif 57.8% atau sebanyak 1022 ulasan, untuk sentimen netral 16.6% atau sebanyak 295 ulasan, dan untuk sentimen negatif 25.4% atau sebanyak 450 ulasan. Hasil pengujian menggunakan SVM menggunakan perbandingan 80% data latih 20% data uji menghasilkan akurasi sebesar 60%. Untuk perbandingan 70% data latih 30% data uji menghasilkan akurasi sebesar 59%. Sedangkan perbandingan 60% data latih 40% data uji menghasilkan akurasi sebesar 58%. Selanjutnya dapat disimpulkan hasil dari Confusion matrix bahwa pengenalan kalimat negatif lebih banyak dikenali sebagai kalimat positif. Berdasarkan hasil Word cloud kata capres, elektabilitas, pemilih, kritis, anis, dan ganjarmenangtotal merupakan kata yang sering muncul.

Keywords

Support Vector Machine, Presidential Candidate, VADER Lexicon, Sentiment

Abstrak

The Indonesian Survey Circle and Indo Barometer surveys put the 3 highest-ranking candidates for president, these figures are Ganjar Pranowo, Anies Baswedan, and Prabowo Subianto. The three figures resulting from the survey are well-known figures. There is a lot of news that talks about this character, as well as the community. The community, whether they support it or not, talks a lot on social media. One of the social media used to comment on this figure is the social media Twitter. With sentiment towards the figure of a presidential candidate, one can find out how the public reacts to this figure, the trend of sentiment towards the figure, and the words that become sentiment. The sentiments given by these Twitter users are mostly in the form of opinions or news, which are very numerous on Twitter. Many automation techniques are used to perform sentiment analysis such as a lexicon-based approach, using machine learning, or a combination of both. Various machine learning algorithms are used, one of which is the Support Vector Machine (SVM). The results of labeling using the VADER Lexicon yielded 57.8% positive sentiment or 1022 reviews, 16.6% neutral sentiment

Indonesian Journal of Computer Science
ISSN 2302-4364 (*print*) dan 2549-7286 (*online*)
Jln. Khatib Sulaiman Dalam, No. 1, Padang, Indonesia, Telp. (0751) 7056199, 7058325
Website: ijcs.stmikindonesia.ac.id | E-mail: ijcs@stmikindonesia.ac.id

or 295 reviews, and 25.4% negative sentiment or 450 reviews. The test results using SVM using a comparison of 80% training data 20% test data produce an accuracy of 60%. For a comparison of 70% training data 30% test data produces an accuracy of 59%. Meanwhile, a comparison of 60% training data and 40% test data yields an accuracy of 58%. Furthermore, it can be concluded from the results of the Confusion matrix that the introduction of negative sentences is more widely recognized as positive sentences. Based on the results of the word cloud, the words presidential candidate, electability, voter, critical, anis, and total win reward are the words that appear frequently.

A. Pendahuluan

Memasuki tahun 2024, Indonesia masuk ke tahun politik, dimana para peserta pemilihan umum mulai mempersiapkan diri untuk menghadapi pesta demokrasi pemilihan umum presiden yang akan berlangsung pada tahun 2024. Berbagai lembaga telah melakukan survei lapangan untuk mengetahui elektabilitas terhadap seorang figur bakal calon presiden. Tiga lembaga survei atau lembaga politik telah melakukan survei untuk menentukan elektabilitas 3 besar calon presiden 2024. Hasil survei Lingkaran Survei Indonesia dan Indo Barometer menempatkan 3 figur bakal calon presiden tertinggi, figur tersebut adalah Ganjar Pranowo, Anies Baswedan, dan Prabowo Subianto.

Tiga tokoh yang dihasilkan dari survei tersebut merupakan figur terkenal. Banyak berita yang membicarakan tokoh tersebut, begitu juga dengan masyarakat. Masyarakat baik yang mendukung ataupun tidak, banyak membicarakan pada media sosial. Salah satu media sosial yang digunakan untuk beropini figur tersebut adalah media sosial Twitter. Pada media sosial Twitter, pengguna dapat dapat mem-posting, membaca, dan mengomentari apa saja yang dituliskan oleh pengguna, sehingga pengguna dapat membaca sebuah informasi kemudian memberikan sebuah sentimen atau opini dalam bentuk positif atau negatif terhadap figur bakal calon presiden 2024. Media sosial Twitter banyak digunakan oleh pengguna internet dalam hal untuk mengikuti perkembangan berita dari pada media sosial lainnya meskipun Twitter kalah dalam hal jumlah pengguna aktif. Jumlah penggunaan Twitter untuk mengikuti berita diperkuat oleh hasil publikasi Pew Research Center dalam berita BloombergGadfly yang menyatakan sebanyak 70% lebih pengguna menggunakan Twitter untuk mengikuti pemerintahan atau politik dibandingkan Facebook yang sebesar 60%.

Sentimen adalah pendapat atau pandangan yang didasarkan pada perasaan yang berlebih-lebihan terhadap sesuatu. Sedangkan analisis sentimen adalah proses identifikasi otomatis apakah teks yang dihasilkan oleh pengguna mengekspresikan pendapat positif, negatif, dan netral terhadap suatu objek (produk, orang, topik, acara, dan lain lain).

Dengan sentimen terhadap figur bakal calon presiden, dapat mengetahui bagaimana masyarakat menyikapi terhadap figur tersebut, trend sentimen terhadap figur, dan kata-kata yang menjadi sentimen. Sentimen-sentimen yang diberikan oleh pengguna Twitter tersebut banyak berupa opini ataupun berita yang jumlahnya sangat banyak di Twitter. Untuk menganalisa data tweet yang jumlahnya banyak dibutuhkan suatu teknik otomasi, sehingga sentimen tersebut dapat diklasifikasi apakah termasuk sentimen positif atau sentimen negatif secara cepat tanpa memeriksa secara manual. Proses mengolah data sentimen yang sangat banyak biasanya disebut analisis sentimen atau opinion mining.

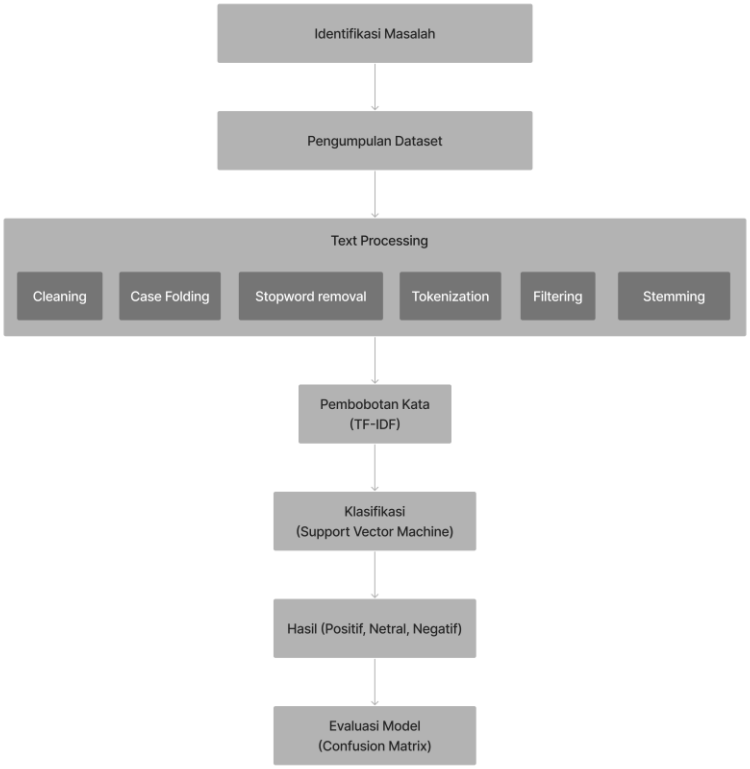
Banyak teknik otomasi yang digunakan untuk melakukan analisis sentimen seperti pendekatan berbasis leksikon, menggunakan machine learning, atau gabungan keduanya. Algoritma machine learning yang digunakan beragam seperti Maximum Entropy (ME), Naïve Bayes (NB), dan Support Vector Machine (SVM). Berbagai macam penelitian telah dilakukan untuk membandingkan kemampuan algoritma tersebut. Pada penelitian Chandani dan Wahono menghasilkan akurasi algoritma SVM sebesar 81.10% dibandingkan NB 74.00%. Pada penelitian

Saraswati algortima SVM menunjukkan akurasi lebih baik daripada NB dengan nilai masing-masing 75.06% dan 74.39%.

Berdasarkan uraian di atas penulis bermaksud untuk melakukan penelitian tentang "Support Vector Machine untuk Sentiment Analysis Bakal Calon Presiden Republik Indonesia 2024". Penelitian ini diharapkan mampu menghasilkan klasifikasi sentimen publik terhadapt Bakal Calon Presiden 2024.

B. Metode Penelitian

Proses melakukan sebuah penelitian data dan informasi yang bersifat objektif yang akan digunakan sebagai titik acuan dalam penelitian, dengan adanya data-data tersebut di harapkan penelitian yang di hasilkan adalah penelitian yang berkualitas. Proses dalam melakukan penelitian ini digambarkan sebagai berikut:



Gambar 1. Tahapan Metode Penelitian

Tahapan yang dilakukan dapat dijelaskan sebagai berikut:

1. Identifikasi Masalah
   Pembuatan sistem ini diawali dengan menentukan masalah, dalam peneltian ini adalah mengklasifikasikan opini masyarakat terhadap figur bakal calon presiden 2024 sehingga dapat diketahui pola-pola tertentu yang dijadikan pengetahuan baru.
2. Pengumpulan Dataset
   Dataset yang digunakan pada penelitian ini sebanyak 2000 data tweet yang diambil mulai dari 3 Oktober 2022 hingga 28 Februari 2023 yang diperoleh dari media sosial Twitter. Berikut adalah bentuk tampilan data tweet yang diperoleh.

Topic	Topic keywords
1	ganjarpranowo, ganjar, ganjar_pranowo, ganjarkita, ganjaruntukindonesia, ganjarpilihanrakyat, ganjarpranowofor2024, sahabatganjar, ganjarhebat, relawanganjar
2	ani, rt, baswedan, pemimpin, prabowo, yg, 2024, indonesia, presiden, relawanani
3	prabowo, subianto, pilpres2024, dekade08, mendingprabowo, terusmajubersamaprabowo, surabaya, 2024, presiden, ani
4	ganjar, pranowo, ganjarpilihanjokowi, dukung, saga_jatim, 2024, ani, rt, ganjaramanah, ganjaranapp
5	pranowo, capres2024, ganjarpilihanjokowi, ganjar, saga_jatim, ganjaranapp, 2024, dukung, maju, indonesia
6	pemimpin, yg, presiden, rt, ani, 2024, cerda, rahmansandr, setuju, kalangan
7	rt, capres2024, ganjarpilihanjokowi, pemimpin, pranowo, yg, 2024, kalangan, cerda, dinilai
8	2024, presiden, capres2024, ganjarpilihanjokowi, dukung, indonesia, relawanganjar, ganjaranapp, setuju, cerda
9	capres2024, indonesia, rt, ganjarpilihanjokowi, saga_jatim, aniesdidukungkhilafah, aniesnasdemout, tenggelamkananiesbaswedan, tenggelamkanpartainasdem, relawanganjar
10	pemimpin, helmifelis_, turunkan, masuk, fakta, tuhan, kategori, indonesi, jeniu, indonesia

Gambar 2. Data Tweet Bakal Calon Presiden 2024

3. *Preprocessing*
   Tahap *preprocessing* diartikan sebagai tahap awal untuk untuk membersihkan kata-kata yang tidak perlu atau kata-kata yang tidak memiliki makna. Proses keseluruhan dilakukan menggunakan program Python3, sehingga dilakukan secara otomatis. *Preprocessing* yang akan dilakukan mencakup berbagai aktivitas, diantaranya adalah:
   1. Pembersihan data (*data cleaning*)
      Menghilangkan karakter yang tidak diinginkan seperti tanda baca, url, emoji, angka, atau karakter khusus dari teks.
   2. *Casefolding*
      Mengubah teks menjadi huruf kecil atau huruf besar yang bertujuan untuk menghilangkan perbedaan yang ditimbulkan oleh perbedaan penulisan huruf besar dan kecil dalam dokumen teks.
   3. *Tokenisasi (tokenization)*
      Memecah teks menjadi kata-kata atau frase yang lebih kecil.
   4. *Stopword removal*
      Menghilangkan kata-kata yang tidak memiliki makna yang signifikan seperti “apakah”, “ke”, “luar”, ”biasanya”, dll.
   5. *Filtering*
      Menghilangkan kata-kata yang tidak sesuai dengan topik yang dianalisis, hal ini akan membuat data lebih bersih dan akurat sebelum digunakan dalam proses analisis.
   6. *Stemming*
      Menghilangkan imbuhan dari kata-kata untuk mengurangi variasi kata yang sama.
4. Pembobotan Kata (TF-IDF)
   Setelah data sudah dibersihkan dan selesai dari tahapan preprocessing maka selanjutnya semua datadikonversikan kedalam bentuk angka agar memiliki nilai bobotnya sehingga dapat di proses oleh algoritma klasifikasi. Nilai bobot ini ditentukan berdasarkan konteks atau makna dari kata tersebut dalam teks. Pembobotan kata pada penelitian ini menggunakan metode TF-IDF.
5. Klasifikasi Data
   Sebelum melakukan klasifikasi data, perlu dilakukannya beberapa tahapan seperti *spliting data,* pembangunan algoritma SVM dan melatih algoritma tersebut untuk melakukan evaluasi. *Splitting data* digunakan membagi data

menjadi dua bagian, yaitu data latih (*training*) dan data uji (*testing*). Pada penelitian ini, penulis melakukan empat percobaan.

1. Percobaan pertama, 60% data digunakan sebagai data latih dan 40% sebagai data uji.
2. Percobaan kedua, 70% data sebagai data latih dan 30% sebagai data uji.
3. Percobaan ketiga, 80% data sebagai data latih dan 20% sebagai data uji.

Tahap selanjutnya adalah melakukan evaluasi terhadap performa dari algoritma SVM, performa akan diuji berdasarkan metode *confusion matrix* dengan menggunakan data uji.

6. Hasil
   Setelah data diproses dengan algoritma SVM maka akan di peroleh hasil berupa data yang menunjukkan kategori postitif, negatif, dan netral. Pada saat klasifikasi, algoritma akan mencari probabilitas tertinggi dari semua kategori dokumen yang diujikan. Selanjutnya data divisualisasikan dalam bentuk diagram agar menghasilkan grafik dari masing-masing kategori.
7. Evaluasi Model *Confusion Matrix*
   Cara mengetahui performa dari metode SVM, maka dilakukan pengujian terhadap model. Hasil klasifikasi akan ditampilkan dalam bentuk *confusion matrix*. Nilai akurasi yang didapatkan dihitung berdasarkan jumlah nilai dari diagonal *confusion matrix* dibagi dengan jumlah seluruh data. Selanjutnya evaluasi dilakukan untuk mengetahui kinerja model. Evaluasi model dilakukan dengan cara melihat tingkat akurasi metode melalui *confusion matrix* dan tabel akurasi serta presisi untuk tiap model. Setelah data uji diujikan, maka akan menghasilkan daftar kelas-kelas dari data uji, sebut saja prediksi kelas. Kemudian prediksi kelas dibandingkan dengan kelas yang sebenarnya dari data uji yang disembunyikan sebelumnya. Sehingga dapat dilihat dan dihitung nilai *accuration*, *precision*, *recall*, dan *f1-score*. Setelah diketahui hasil, selanjutnya untuk performa metode klasifikasi dari setiap kelasnya dapat dilihat melalui nilai presisi, *recall*, dan *f1-score* pada setiap kelasnya. Hasil nilai presisi, *recall*, dan *f1-score* memiliki nilai sebesar 0-1. Semakin tinggi nilainya maka semakin baik. Gambaran hasil dari proses evaluasi model dapat dilihat pada Tabel 1.

Tabel 1. Gambaran Nilai Presisi, *Recall, F1-Score*

Jenis Klasifikasi	presisi	*recall*	*f1-score*
Positif	?	?	?
Netral	?	?	?
Negatif	?	?	?

Sehingga disimpulkan hasil dari evaluasi model dapat dilihat nilai presisi, *recall*, dan *f1-socre* di setiap kelasnya.

C. Hasil dan Pembahasan

1. Pengumpulan Data

Salah satu keunggulan Python adalah mendukung banyak *open-source library*. Ada banyak library Python yang dapat digunakan. Pada gambar 3 dibawah

ini adalah library yang diperlukan dalam pemrosesan sentimen analis bakal calon presiden 2024.

```
#masukan modul yang dibutuhkan
import pandas as pd
import numpy as np
import nltk
import string
import re
```

Gambar 3. Library Python

Berikutnya dilakukan proses load dataset yang sudah diperoleh dari twitter. Pada gambar 4 dibawah ini adalah proses yang dilakukan untuk membaca dataset yang menggunakan extensi .csv.

```
#masukan data kembali yang akan diproses
def load_data():
    data_tweets = pd.read_csv('crawling_capres.csv')
    return data_tweets

tweets = load_data()
tweets.head() #tampilkan bagian atas saja
```

Gambar 4. Load Dataset CSV

2. Text Processing Data

Tahapan untuk membersihkan data dari fitur-fitur yang tidak diinginkan. Tahapan yang dilakukan: *remove user, cleaning, case folding, tokenisasi, stopword removal,* dan *stemming.*

2.1 Remove User

Langkah pertama tahapan text preprocessing adalah melakukan remove user dari tadaset. Pada gambar 5 dibawah ini adalah proses yang dilakukan.

```
#menghapus username dalam tweet
def remove_pattern(Text, pattern):
    r = re.findall(pattern, str(Text))
    for i in r:
        Text = re.sub(i, '', str(Text))
    return Text
tweets['remove_user'] = np.vectorize(remove_pattern)(tweets['Text'], "@[\w]*")
```

Gambar 5. Load Dataset CSV

2.2 Cleaning

Data cleaning merupakan suatu proses mendeteksi dan memperbaiki suatu record yang ‘corrupt’ atau tidak akurat berdasarkan sebuah record set, tabel, atau database. Selain itu, data cleansing juga berguna untuk mengidentifikasi bagian data mana yang tidak lengkap, tidak tepat, tidak akurat atau tidak relevan, yang selanjutnya untuk data-data “kotor” tersebut akan diganti, dimodifikasi, atau dihapus. Pada gambar 6 dibawah ini merupakan proses program Python yang dilakukan.

```
#cleaning
def cleaning(Text):
    Text = re.sub(r'\$\w*', '', Text)
    Text = re.sub(r'^rt[\s]+', '', Text)
    Text = re.sub('((www\.[^\s]+)|(https?://[^\s]+))', ' ', Text)
    Text = re.sub('&quot;'," ", Text)
    Text = re.sub(r"\d+", " ", str(Text))
    Text = re.sub(r"\b[a-zA-Z]\b", "", str(Text))
    Text = re.sub(r"[^\w\s]", " ", str(Text))
    Text = re.sub(r'(.)\1+', r'\1\1', Text)
    Text = re.sub(r"\s+", " ", str(Text))
    Text = re.sub(r'#', '', Text)
    Text = re.sub(r'[^a-zA-z0-9]', ' ', str(Text))
    Text = re.sub(r'\b\w{1,2}\b', '', Text)
    Text = re.sub(r'\s\s+', ' ', Text)
    Text = re.sub(r'^RT[\s]+', '', Text)
    Text = re.sub(r'^b[\s]+', '', Text)
    Text = re.sub(r'^link[\s]+', '', Text)
    return Text

tweets['cleaning'] = tweets['remove_user'].apply(cleaning)
```

Gambar 6. Proses Data Cleaning

2.3 *Case Folding*

Case Folding yang dilakukan adalah perubahan huruf kapital menjadi huruf kecil, jadi semua teks yang mengandung huruf kapital akan diubah menjadi huruf kecil. Tujuan dari Case Folding ini adalah agar kata-kata yang sama tidak terdeteksi berbeda hanya karena perbedaan terdapat huruf kapital. Pada gambar 7 dibawah ini merupakan proses program Python yang dilakukan.

```
#case folding - ubah jadi huruf kecil
tweets['case_folding'] = tweets['cleaning'].str.lower()
```

Gambar 7. Proses Case Folding

2.4 *Tokenisasi*

Tokenizing adalah operasi memisahkan teks menjadi potongan-potongan berupa token, bisa berupa potongan huruf, kata, atau kalimat, sebelum dianalisis lebih lanjut.. Pada gambar 8 dibawah ini merupakan proses program Python yang dilakukan.

```
#tokenisasi - membagi kalimat jadi perkata (dipisah)
nltk.download('punkt')
from nltk.tokenize import word_tokenize

def word_tokenize_wrapper(Text):
    return word_tokenize(Text)

tweets['tokenisasi'] = tweets['case_folding'].apply(lambda x: word_tokenize_wrapper(x.lower()))
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\Firman\AppData\Roaming\nltk_data...
[nltk_data]   Package punkt is already up-to-date!
```

Gambar 8. Proses Tokenisasi

2.5 *Stopword Removal*

Stopword adalah kata umum yang biasanya muncul dalam jumlah besar dan dianggap tidak memiliki makna. Pada gambar 9 dibawah ini merupakan proses program Python yang dilakukan.

```
#stopword removal - menghapus kata sesuai dengan kamus indonesia
nltk.download('stopwords')
from nltk.corpus import stopwords

list_stopwords = stopwords.words('indonesian')

list_stopwords.extend(['yg', 'dg', 'rt', 'dgn', 'ny', 'd', 'klo',
                       'kalo', 'amp', 'biar', 'bikin', 'bilang',
                       'gak', 'ga', 'krn', 'nya', 'nih', 'sih',
                       'si', 'tau', 'tdk', 'tuh', 'utk', 'ya',
                       'jd', 'jgn', 'sdh', 'aja', 'n', 't',
                       'nyg', 'hehe', 'pen', 'u', 'nan', 'loh', 'rt',
                       '&', 'yah', 'no', 'je', 'om', 'pru', 'sch',
                       'injirrr', 'ah', 'oena', 'bu', 'eh', 'n', 'anjir'])

list_stopwords = set(list_stopwords)

def stopwords_removal(Text):
  return [word for word in Text if word not in list_stopwords]

tweets['stopword_removal'] = tweets['tokenisasi'].apply(stopwords_removal)
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\Firman\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Gambar 9. Proses Stopword

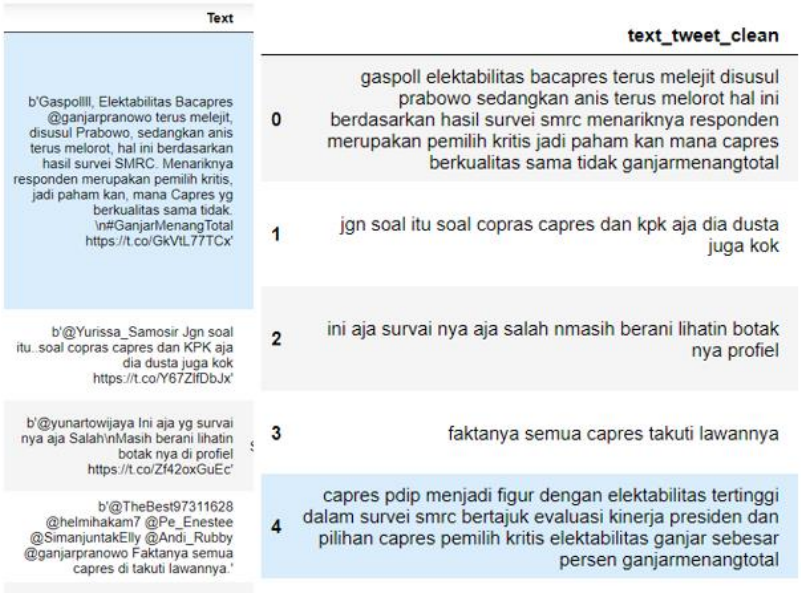
2.6 Stemming

Stemming adalah proses menghilangkan infleksi kata ke bentuk dasarnya, namun bentuk dasar tersebut tidak berarti sama dengan akar kata (*root word*). Pada gambar 10 dibawah ini merupakan proses program Python yang dilakukan.

```
#stemming - menghapus imbuhan
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
from Sastrawi.StopWordRemover.StopWordRemoverFactory import StopWordRemoverFactory
import swifter

#buat stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()

#stemmed wrapper
def stemmed_wrapper(term):
  return stemmer.stem(term)

term_dict = {}

for Text in tweets['stopword_removal']:
  for term in Text:
    if term not in term_dict:
      term_dict[term] = ' '

print(len(term_dict))
print("------------------------")

for term in term_dict:
    term_dict[term] = stemmed_wrapper(term)
    print(term,":" ,term_dict[term])

print(term_dict)
print("------------------------")

#memmulai stemming
def apply_stemmed_term(Text):
  return [term_dict[term] for term in Text]

tweets['stemming'] = tweets['stopword_removal'].swifter.apply(apply_stemmed_term)
```

Gambar 10. Proses Stemming

Setelah semua proses preprocessing dijalankan terhadap semua data, maka hasil dari preprocessing disimpan menjadi suatu file baru yang nantinya akan dijadikan sebagai dataset dalam proses pengklasifikasian. Adapun hasil dari proses text preprocessing pada Gambar 11 di bawah.



Text		text_tweet_clean
b'Gaspolll, Elektabilitas Bacapres @ganjarpranowo terus melejit, disusul Prabowo, sedangkan anis terus melorot, hal ini berdasarkan hasil survei SMRC. Menariknya responden merupakan pemilih kritis, jadi paham kan, mana Capres yg berkualitas sama tidak. \n#GanjarMenangTotal https://t.co/GkVtL77TCx'	0	gaspoll elektabilitas bacapres terus melejit disusul prabowo sedangkan anis terus melorot hal ini berdasarkan hasil survei smrc menariknya responden merupakan pemilih kritis jadi paham kan mana capres berkualitas sama tidak ganjarmenangtotal
b'@Yurissa_Samosir Jgn soal itu..soal copras capres dan KPK aja dia dusta juga kok https://t.co/Y67ZIfDbJx'	1	jgn soal itu soal copras capres dan kpk aja dia dusta juga kok
b'@yunartowijaya Ini aja yg survai nya aja Salah\nMasih berani lihatin botak nya di profiel https://t.co/Zf42oxGuEc'	2	ini aja survai nya aja salah nmasih berani lihatin botak nya profiel
b'@TheBest97311628 @helmihakam7 @Pe_Enestee @SimanjuntakElly @Andi_Rubby @ganjarpranowo Faktanya semua capres di takuti lawannya.'	3	faktanya semua capres takuti lawannya
	4	capres pdip menjadi figur dengan elektabilitas tertinggi dalam survei smrc bertajuk evaluasi kinerja presiden dan pilihan capres pemilih kritis elektabilitas ganjar sebesar persen ganjarmenangtotal

Gambar 11. Hasil Text Preprocessing

3. Pelabelan Data

Pelabelan data merupakan proses untuk mengkategorikan kumpulan data ke dalam kelompok tertentu. Proses pelabelan menggunakan vader lexicon. VADER (*Valence Aware Dictionary and Sentiment Reasoner*) yaitu pendekatan yang memungkinkan untuk mengklasifikasikan informasi teks ke dalam beberapa kategori sentimen yaitu negatif, positif, dan netral. Kemampuan klasifikasi dilakukan VADER dengan cara memberikan nilai pada setiap kata dalam teks. Pada gambar 12 dibawah ini adalah proses pada program Python yang dilakukan.

```
#import modul yang dibuthkan labelin vader
import nltk
nltk.download('vader_lexicon')
from nltk.sentiment.vader import SentimentIntensityAnalyzer
sid = SentimentIntensityAnalyzer()

import pandas as pd
```

```
[nltk_data] Downloading package vader_lexicon to
[nltk_data]     C:\Users\Firman\AppData\Roaming\nltk_data...
[nltk_data]   Package vader_lexicon is already up-to-date!
```

Gambar 12. Proses Vader Lexicon

Setelah proses diatas dilakukan maka diperoleh hasil pelabelan data seperti terlihat pada gambar 13 dibawah ini.

	text_tweet_clean	scores	compound	sentimen
0	gaspoll elektabilitas bacapres terus melejit disusul prabowo sedangkan anis terus melorot hal ini berdasarkan hasil survei smrc menariknya responden merupakan pemilih kritis jadi paham kan mana capres berkualitas sama tidak ganjarmenangtotal	{'neg': 0.101, 'neu': 0.848, 'pos': 0.051, 'compound': -0.3964}	-0.3964	negatif
1	jgn soal itu soal copras capres dan kpk aja dia dusta juga kok	{'neg': 0.236, 'neu': 0.764, 'pos': 0.0, 'compound': -0.5267}	-0.5267	negatif
2	ini aja survai nya aja salah nmasih berani lihatin botak nya profiel	{'neg': 0.096, 'neu': 0.703, 'pos': 0.201, 'compound': 0.5023}	0.5023	positif
3	faktanya semua capres takuti lawannya	{'neg': 0.0, 'neu': 1.0, 'pos': 0.0, 'compound': 0.0}	0.0000	neutral
4	capres pdip menjadi figur dengan elektabilitas tertinggi dalam survei smrc bertajuk evaluasi kinerja presiden dan pilihan capres pemilih kritis elektabilitas ganjar sebesar persen ganjarmenangtotal	{'neg': 0.0, 'neu': 0.92, 'pos': 0.08, 'compound': 0.2732}	0.2732	positif

Gambar 13. Hasil Pelabelan Tweet dengan Vadel Lexicon

4. Pembobotan Kata TF-IDF

Tahap selanjutnya adalah pembobotan kata yang ada pada dokumen. Pada penelitian ini menggunakan library sklearn. Modul yang digunakan adalah *CountVectorizer* dan *Tf-Idf vectorizer*. Dapat dilihat pada gambar berikut.

```
from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = TfidfVectorizer()
X = vectorizer.fit_transform(df['text_tweet_clean'])
```

Gambar 14. Pembobotan Dokumen

5. Klasifikasi SVM

Setelah melakukan pembobotan dokumen menggunakan TF-IDF tahap berikutnya adalah melakukan klasifikasi. Klasifikasi yang digunakan yaitu SVM. Pada tahap klasifikasi akan dilakukan perbandingan data latih dan data uji menggunakan perbandingan 60:40, 70:30, dan 80:20 di split data. Dapat dilihat pada gambar berikut.

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.4, random_state=42)
```

```
#Support Vector Machine
from sklearn.svm import SVC

svm_model = grid_search.best_estimator_
svm_model.fit(X_train, y_train)
predicted_svm = svm_model.predict(X_test)

from sklearn.metrics import classification_report
print(classification_report(y_test, predicted_svm))
```

```
              precision    recall  f1-score   support

          -1       0.48      0.33      0.39       191
           0       0.45      0.19      0.27       125
           1       0.61      0.82      0.70       391

    accuracy                           0.58       707
   macro avg       0.51      0.45      0.45       707
weighted avg       0.55      0.58      0.54       707
```

Gambar 15. Perbandingan 60:40

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, random_state=42)
```

```
#Support Vector Machine
from sklearn.svm import SVC

svm_model = grid_search.best_estimator_
svm_model.fit(X_train, y_train)
predicted_svm = svm_model.predict(X_test)

from sklearn.metrics import classification_report
print(classification_report(y_test, predicted_svm))
```

```
              precision    recall  f1-score   support

          -1       0.48      0.43      0.45       148
           0       0.47      0.20      0.28        95
           1       0.64      0.80      0.71       288

    accuracy                           0.59       531
   macro avg       0.53      0.47      0.48       531
weighted avg       0.57      0.59      0.56       531
```

Gambar 16. Perbandingan 70:30

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42)
```

```
#Support Vector Machine
from sklearn.svm import SVC

svm_model = grid_search.best_estimator_
svm_model.fit(X_train, y_train)
predicted_svm = svm_model.predict(X_test)

from sklearn.metrics import classification_report
print(classification_report(y_test, predicted_svm))
```

```
              precision    recall  f1-score   support

          -1       0.50      0.47      0.49        97
           0       0.53      0.24      0.33        66
           1       0.65      0.79      0.71       191

    accuracy                           0.60       354
   macro avg       0.56      0.50      0.51       354
weighted avg       0.59      0.60      0.58       354
```

Gambar 17. Perbandingan 80:20

Untuk proses klasifikasi ini menggunakan model SVM dari library *sklearn* dengan memanggil masing-masing modelnya. Dari masing-masing model tersebut akan diterapkan ke data latih yang sudah terbentuk. Kemudian akan memprediksi

sentimen dari ulasan pengguna aplikasi Sausage Man yang akan menjadi patokan data uji.

6. Evaluasi Confusion Matrix

Setelah melakukan klasifikasi menggunakan SVM tahap terakhir yaitu evaluasi menggunakan *Confusion matrix* untuk mengetahui akurasi, presisi, recall. Hasil pengujian dapat dilihat pada tabel dibawah ini.

Tabel2. Hasil Pengujian

Jumlah Kelas	Data Latih	Data Uji	Akurasi	Recall	Precision
3	60%	40%	58%	33%	48%
3	70%	30%	59%	43%	48%
3	80%	20%	60%	47%	50%

Dari pengujian yang sudah dilakukan dengan menghitung akurasi, recall, dan precision maka analisis dapat disimpulkan.

1. Pada tabel 7 dapat dilihat dari nilai akurasi menggunakan 80% data latih 20% data uji memiliki akurasi tertinggi 60%. Berikutnya pada pengujian 70% data latih 30% data uji memiliki akurasi 59%. Sedangkan pada pengujian 60% data latih 40% data uji memiliki akurasi terendah 58%.
2. Untuk nilai akurasi tertinggi analisis sentimen dari pengujian yang sudah dilakukan terdapat pada kelas positif.

Berikut ini Hasil word cloud positif, netral, dan negatif pada analisis sentimen ini dapat dilihat pada gambar berikut.

Gambar 18. Word cloud Sentimen Bakal Calon Presiden 2024

D. Simpulan

Berdasarkan penelitian dan pembahasan yang sudah dilakukan yakni Klasifikasi Sentiment Analysis Bakal Calon Presiden 2024 dengan SVM dapat disimpulkan bahwa hasil pelabelan menggunakan VADER Lexicon di dapatkan hasil sentimen positif 57.8% atau sebanyak 1022 ulasan, untuk sentimen netral 16.6% atau sebanyak 295 ulasan, dan untuk sentimen negatif 25.4% atau sebanyak 450 ulasan. Hasil pengujian menggunakan SVM menggunakan perbandingan 80% data latih 20% data uji menghasilkan akurasi sebesar 60%. Untuk perbandingan 70% data latih 30% data uji menghasilkan akurasi sebesar 59%. Sedangkan perbandingan 60% data latih 40% data uji menghasilkan akurasi sebesar 58%. Selanjutnya dapat disimpulkan hasil dari Confusion matrix bahwa pengenalan kalimat negatif lebih banyak dikenali sebagai kalimat positif. Berdasarkan hasil Word cloud kata capres, elektabilitas, pemilih, kritis, anis, dan ganjarmenangtotal merupakan kata yang sering muncul.

E. Ucapan Terima Kasih

Ucapan terima kasih kepada bapak/ibu dosen di STMIK Amik Riau dan pihak yang sudah membantu dalam penelitian ini.

F. Referensi

[1] Afdhal, I., Kurniawan, R., Iskandar, I., Salambue, R., Budianita, E., & Syafria, F. (2022). Penerapan Algoritma Random Forest Untuk Analisis Sentimen Komentar Di YouTube Tentang Islamofobia. *Penerapan Algoritma Random Forest Untuk Analisis Sentimen Komentar Di YouTube Tentang Islamofobia*, *5*(1), 122–130.

[2] Akbar, F., & Rahmaddeni, R. (2022). Komparasi Algoritma Machine Learning Untuk Memprediksi Penyakit Alzheimer. *Jurnal Komputer Terapan Vol*, *8*(2), 236–245.

[3] Vinita Chandani, Romi Satria Wahono, Purwanto. Komparasi Algoritma Klasifikasi Machine Learning Dan Feature Selection pada Analisis Sentimen Review Film. Journal of Intelligent Systems, Vol. 1, No. 1, February 2015.

[4] Muhammad Raihan Fais Sya' bani, dkk. 2022. Analisis Sentimen Terhadap Bakal Calon Presiden 2024 dengan Algoritma Naïve Bayes. JURIKOM (Jurnal Riset Komputer), Vol. 9 No. 2.

[5] Muhammad Harris Syafa'at, dkk. 2021. SVM Untuk Sentiment Analysis Calon Kepala Daerah Berdasar Data Komentar Video Debat Pilkada Di Youtube. Antivirus: Jurnal Ilmiah Teknik Informatika Vol. 15 No. 2

[6] Rahmaddeni, dkk. 2022. Comparison of support vector machine and XGBSVM in analyzing public opinion on Covid-19 vaccination. ILKOM Jurnal Ilmiah Vol. 14, No. 1

[7] Azriyan Arham, dkk. 2022. Implementasi Sentiment Analysis Pada Opini Masyarakat Indonesia Di Twitter Terhadap Virus Covid-19 Varian Omicron Dengan Algoritma Naïve Bayes, Decision Tree, Dan Support Vector Machine. Sebatik Vol. 26 No. 2.

[8] Rizki Anom Raharjo, dkk. 2022. Perbandingan Metode Naïve Bayes Classifier Dan Support Vector Machine Pada Kasus Analisis Sentimen Terhadap Data Vaksin Covid-19 Di Twitter. Jurnal Elektronika dan Komputer Vol 15 No 2.
[9] Sabily, A.F., Adikara, P.P. & Fauzi, M.A., 2019. Analisis Sentimen Pemilihan Presiden 2019 pada Twitter menggunakan Metode Maximum Entropy. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 3, pp.4204-09.
[10] Gunawan, F., Fauzi, M. A. & Adikara, P. P., 2017. Analisis Sentimen Pada Ulasan Aplikasi Mobile Menggunakan Naive Bayes. Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 1(10), pp. 1082-1088.
[11] Pravina, A.M., Cholissodin, I. & Adikara, P.P. (2019). Analisis Sentimen Tentang Opini Maskapai Penerbangan pada Dokumen Twitter Menggunakan Algoritme Support Vector Machine (SVM). Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, 3(3), 2789-2797.
[12] Nurrohmat, M. A., & SN, A. (2019). Sentiment Analysis of Novel Review Using Long Short-Term Memory Method. IJCCS (Indonesian Journal of Computing and Cybernetics Systems), 13(3), 209. https://doi.org/10.22146/ijccs.41236.
[13] Firmansyah, M. R., Ilyas, R., Kasyidi, F., Informatika, J. T., Jenderal, U., & Yani, A. (2020). Klasifikasi Kalimat Ilmiah Menggunakan Recurrent Neural Network. 26–27.
[14] Muchammad Shiddieqy Hadna, N., Insap Santosa, P., & Wahyu Winarno, W. (2016). Studi Literatur Tentang Perbandingan Metode Untuk Proses Analisis Sentimen Di Twitter. Seminar Nasional Teknologi Informasi Dan Komunikasi, 2016(Sentika), 2089–9815.
[15] M. Rangga, A. Nasution, dan M. Hayaty, "Perbandingan Akurasi dan Waktu Proses Algoritma K-NN dan SVM dalam Analisis Sentimen Twitter," vol. 6, no. 2, pp. 226–235, 2019.
[16] Neneng., Adi. K., Isnanto. R. R., 2016. Support Vector Machine Untuk Klasifikasi Citra Jenis Daging Berdasarkan Tekstur Menggunakan Ekstraksi Ciri Gray Level Co-Occurrence Matrices (GLCM), Universitas Diponogoro.
[17] A.Nurdin. (2020). Teori Komunikasi Interpersonal disertai Contoh Fenomena Praktis Edisi Pertama. Jakarta: Kencana
[18] Selivanov, D. (2020) GloVe Word Embeddings. Available at: https://cran.r-project.org/web/packages/text2vec/vignettes/glove.html (Accessed: 21 Maret 2023).
[19] P. Y. Saputra "Implementasi Teknik Crawling Untuk Pengumpulan Data Dari Media Sosial Twitter" Jurnal Dinamika Dotcom, Volume 8 Nomor 2, Juli 2017.
[20] P. P. A. Arsya Monica Pravina, Imam Cholissodin, "Analisis Sentimen Tentang Opini Maskapai Penerbangan pada Dokumen Twitter Menggunakan Algoritme Support Vector Machine (SVM)," J. Pengemb. Teknol. Inf. dan Ilmu Komput., vol. 3, no. 3, pp. 2789–2797, 2019, [Online]. Available: http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/4793.